**MY (ABILITY) POWER IS OVER 9000!**

[**league.rprevidi.com**](http://league.rprevidi.com)

*RIOT API CHALLENGE 2.0  
Entry’s Category #2: AP items changes between patch 5.11 and 5.14  
  
State: GA   
Country: USA   
Summoner ID: 38762692 aka “Merricat” aka Raimondo Previdi  
Server Region: NA   
  
State: GA   
Country: USA   
Summoner ID: 38905274 aka “Osbourne Cox” aka Jessalyn Sbaratta  
Server Region: NA*

**Concept**

Ideally my team will look at AP champions (mainly mages), existing AP items and Runeglaive, and focus on their relation to victories/pick rate/damage, which should highlight the meaning and the impact of the patch changes (5.11 to 5.14).

**Goal**

Compare AP item usage between patches 5.11 and 5.14 by showing how item changes affected gameplay.

**The Project (in reality)**

Premise: having only been able to work on this challenge in a limited amount of time (we actually started on August 19th), we did not end up exploring as much as we would have liked to.

The project itself is the website [league.rprevidi.com](http://league.rprevidi.com).

**Our approach**

Instead of diving into specific topics (such as Runeglaive and other AP-specific items), we decided to focus our attention on three main categories: Items, Champions, and Roles.

We realized that showing only changes related to AP items would in fact be less informative. This way, by comparing *all* items we can clearly see a pattern of how AP items also affected *other* items, and not just themselves. For instance, while Luden’s Echo might have been picked fewer times in 5.14 compared to Rabadon’s Deathcap, the pick rate of AD items such as Blade of the Ruined King and Infinity Edge might have gone up because of the AP changes.

For Champions and Roles is the same deal. We wanted to show Pick and Win Rate stats between the two patches. We felt like these two categories specifically might be more influenced by the patch changes, especially given the amount of AP champions that are now able to jungle thanks to Runeglaive.

On top of displaying the relevant information regarding AP items, we thought it would be interesting to see how patch changes affected players in other queue types (Ranked Games vs Normals), ELOs and even Regions (and maybe learn a thing or two from them). Apparently Korean players love those health potions!

**Our roles**

Since at first we were not able to do any practical work on the project, we spent that time designing the app both esthetically and functionally. When we got back home to our work stations, my teammate Jess and I decided to split the work into art and programming (our fields of study).

Jess is a vfx artist who loves making champion login screens, which is why she wanted to make something that would specifically fit our app. Because of the timing of the API challenge release, we decided to pay tribute to the Arcade theme by creating an app in a similar style. Since we came up with the name of our project first, we decided to theme it around Final Boss Veigar, one of my favorite champions in terms of personality ☺.

Jess went on by creating a motion graphics loop of the Final Boss Veigar splash screen, while I started programming the app.

Given my education in Game Development, I initially wanted to make the app as a Unity web build, which would have made the whole process easier, faster, and cleaner for me. However, I discovered a day into it that web builds are no longer supported on most browsers; I ended up converting it in Web GL, but it would take over 10 seconds to load on a good computer, and it would not open on an older one. So I decided to learn some javaScript, HTML and CSS and get it done that way.

**The Technical Stuff**

Because we were given 400,000 game id’s, I knew I was going to use a database to store all the relevant data. Because of my previous experience with MySQL and only about 12 days to do it, I decided to stick to it. I would create a PHP script that would automate the insertion of data. I therefore created 8 main tables: four for games, and four for participants, each one with different game stats (ranked, normal, patch 5.11, and patch 5.14).

From there I stopped using the main part of the calls to the RIOT API, because we wanted to build a website that would draw data from our server so that we didn’t need to worry about the api-key rate limits when users would start using the app.

After having the 8 giant MySQL tables, I realized that it would take way too long for a query to run in real time from a website, given that our server is actually an old laptop hooked up in the basement. So I decided to create another PHP script that automated the creation and insertion of data of all the possible combinations of tables needed. Indeed, this isn’t great design, but it definitely made each query take a split second as opposed to over 20s. Also, this way I prevented any stress I would put on my server by having more than one concurrent user trying to query it.

Finally, once I had my hundreds of specific tables, I started working on the actual website. I had to learn some javaScript, some HTML, MySQL, and a lot of CSS to make it work.

This is how the program works: the buttons on the HTML page call javaScript functions when clicked, which then call a php file on the server, which retrieves from MySQL the specific table data as JSON; javaScript then decodes it and sends it back to be displayed in a properly formatted HTML table.

**The Website**

I created four different groups of buttons that the user can pick from at the top of the website: category, ranked tiers, queue type, and regions. While the last three act like filters, the ‘category’ group actually changes the subject of the results (items, champions, roles).

The results are sorted initially by Pick Rate (highest to lowest), but can be re-sorted by Win Rate too, both of them from high-to-low or low-to-high.

The numbers in each column displays first the data from patch 5.14, and next to it, in red/green/gray, the delta from patch 5.11.

**The Art Stuff**

Jess started her work by bringing the splash screen image of Final Boss Veigar into Photoshop, where she separated the image into layers based on what would be animated later on. Given her compositing background, she was able to paint behind each layer to account for the movement of each piece, and create a clean plate of the background.

She brought the layers into After Effects to animate Veigar and his environment. She had to study the movements of both Veigar and Final Boss Veigar in order to bring his personality into the animation. With the idea in mind of creating a piece of art that would fit into an arcade world, Jess decided to create particles that would transition to an 8-bit look.

Finally, through the use of plugins such as the Trapcode Suite, she created custom sprites and over 10 particle systems to bring the tiny master of evil to life!

**The Sounds**

Once we inserted into the background of the website the Motion Graphics that Jess made, we really wanted the page to come alive some more, so I downloaded Audacity and figured out a way to cut together two unique quotes from the Final Boss Veigar skin, and mix it with the newest song that was released with the login screen of Arcade Riven and Battle Boss Blitzcrank. Now Veigar commands you to watch his Ability Power go over 9000!

**Files on GitHub**

* Website: index.html, /media/, /js/, /fonts/, /css/, /data/, dbCommands.php
* Utility (MySQL): jsonToMysql.php, tableMysql.php, championsIdMap.php, itemsIdMap.php,

**Not on GitHub**

* The “private” folder, containing all the credentials to connect to MySQL ☺

Thank you,

Rai and Jess